Research Paper | Information Technology | E-ISSN No : 2454-9916 | Volume : 6 | Issue : 7 | July 2020

REAL TIME AIR POLLUTION MONITORING SYSTEM USING CLOUD AND IOT

*Dhole Shwetali[1] | Keken Priti[1] | Pawbake Shivani[2] | Rahane Sharayu[3]

[1]Student, Information Technology, Avcoe, Sangamner, India. (*Corresponding Author)

ABSTRACT

This project is to development the centralized cloud-based system using sensors that monitor and analyse air pollution. The collected data by sensor node stored on cloud server that can be accessed through web browser from anywhere using website.

Due to real-time monitoring of environment, quick action can be taken for the presence of pollutant in environment. This project aim is to track of air quality of surrounding and keep the data updated over internet. Readings are taken in real time and through out day.

We developed graphs that easily analysis percentage of pollutant in particular area. The LCD display continuously shows the real time values of gas sensor.

KEYWORDS: Internet of things, Air Pollution, Cloud, Atmosphere Automation, Pollution, Arduino, Sensors.

1. INTRODUCTION:

The quality of air is important for the survival of living beings. It is necessary to monitor air quality and keep it under control for healthy living for all.The air pollution monitoring system is designed to aware the public about the quality of air being affected by different factors like pollutants, toxic gases etc.

IOT Based Air Pollution Monitoring System monitors the Air quality over a web server using Internet and will trigger an alarm when the air quality goes down beyond a certain threshold level, means when there are sufficient amount of harmful gases present in the air like CO, smoke, alcohol, CO2, LPG, dust particle. It show the air quality in PPM on the LCD and as well as on web-page so that it can monitor it very easily. LPG sensor is added in this system which is used mostly in houses. The system will show temperature and humidity. The system can be installed anywhere but mostly in industries and houses where gases are mostly to be found and gives an alert message when the system crosses threshold limit. The air pollution monitoring system consist of three layer architecture.

The data gathered by a system transmitted instantly to a web- based application to enable monitoring real time data. Due implementation of these project it get easier to monitor and control environment using cloud, where all data stored that can be analyzed and seen in real time through web browser with the interface that will be provided to the cloud.

1.1 Sensor Layer:

Sensor network used to detect the presence CO2, CH4, CO, SO2, H2S, dust particle content in the air. The sensor layer includes the electronic hardware circuits and the software components, both for sensor nodes and the gateway node, which assembled using a Node MCU with a wireless expansion module for capturing the data.

These node monitors the sensors, acquires reading, stores the data on cloud and transmits to the gateway.

1.2 Application Layer:

Web services are design and implement using set of protocol and format that are used to process and store data in cloud infrastructure. This information will be published on a Web page so that users would be able to access it through their Web browser. In order to send messages from the Gateway node to the application.

1.3 Client Layer:

This layer provide graphical user interface to visualize the information. This layer consist of Web graphical user interface, providing a visual information about environment parameter in order to allow the communication with the WSN and users.The client can access the data that is being displayed on the dashboard but the client will not be able to do any modification to the data received. We can install this system anywhere to monitor the data. The Web application provides users with the online real time monitoring of the sensor data and also allows the retrieval and control of sensors remotely.

2. LITERATURE SURVEY:

Xing Liu, Orlando [4] presented a comparative study on smart sensors, objects, devices and things in Internet of Things. The authors have also explained the definition and concepts of IoT in various different ways. The differences and similarities between the smart objects, smart things in IoT are presented in tabular form. Marinov, Marin B. et al [3] monitors environmental parameters with amperometric sensors and gas sensors (infrared) using the PIC18F87K22 microcontroller. Sensor nodes are set up in different areas for real time monitoring of environment. The results are displayed on the city map.

Baralis, Elena et al [10] proposes a business intelligence engine (APA). The system is designed to aware the public about the quality of air being affected by different factors like pollutants, toxic gases etc. Analysis of air pollution from different perspectives like meteorological data, pollutants and traffic data using APA is done. The system helps the people to realize their activities impact on deteriorating air quality.

Jha, Mukesh et al [7] presented a system for monitoring the environmental parameters, modeling and manipulating micro- climate of urban areas. The system is implemented for the adaption of efficient urban infrastructure after analyzing the urban micro-climate.

Shete., R. and Agrawal S. [6] provides the framework for monitoring the city environment. Low cost Raspberry pi is used for implanting the system. Parameters like carbon monoxide, carbon dioxide, temperature and pressure are measured but no emphasis is given on particulate matter which left the environment monitoring incomplete.

Phala, Kgoputjo Simon Elvis, Anuj Kumar, [1] focused on development of system Which capable for real time measurements of pollutants in atmosphere using GSM wireless communication model.

Zheng, Kan, Shaohang Zhao, Zhe Yang, Xiong Xiong, and Wei Xiang [2] proposed a System using portable sensor to collect air quality information in real time environment. The data is analysed using Iot cloud.

Nastic, Stefan, Sanjin Sehic, Duc-Hung Le, Hong Linh Truong [8] introduced the concept of IOT units and approaches to design a IOT and cloud systems.

3. SYSTEM DESIGN:

Our System has mainly three modules sensor layer, application layer, cilent layer. Sensor layer Sensor network used to detect the presence CO2, CH4, CO, NH3, C0, LPG, dust particle content in the air. The first sensor layer includes the electronic hard- ware circuits and the software components, both for sensor nodes and the gateway node, which assembled using a Node MCU with a wireless expansion module for cap- turing the data..

MQ-135 is sensitive to smoke, alcohol and CO2. MQ-5 is sensitive to LPG, hydrogen, carbon monoxide. MQ-4 is used to detect methane. MQ-6 detect the LPG, Butane. Libelium provides generic platforms capable of running, interfacing, and data collection from these sensors. The open platforms allow the users to customize the gases they desire to monitor in their facilities. These node monitors the sensors, acquires reading, stores the data on cloud and transmits to the gateway. NodeMCU is an open source IoT platform. It includes _rmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems and hardware which is based on the ESP-12 module. The term "NodeMCU" by default refers to the

_rmware rather than the development kits. The _rmware uses the Lua scripting language. It is based on the eLua project, and built on the Espressif Non- OS SDK for ESP8266. It uses many open source projects, such as lua-cjson and SPIFF [5]. The NodeMCU was able to support the MQTT IoT protocol, using Lua to access the MQTT broker. Another important update was made on 30 Jan 2015, when Devsaurus ported the u8glib to NodeMCU project, enabling NodeMCU to easily drive LCD, Screen, OLED, even VGA displays.

3.1 Verification & Validations:

3.1.1 Verification:

It is the process of evaluating a system or component to determine whether the product of given development phase satisfy the condition imposed at the start of the phase.

- It is a static process.
- It does not involve any code and is human based checking.
- It uses methods like inspections, walk through, desk checking etc.
- It can catch errors that validations cannot.

3.1.2 Validation:

It is the process of evaluating a system or component during or at the end of the development process to determine to determine whether it speci_es the speci_ed requirements. It involves executing the actual software. it is a computer based testing process.

- It is a dynamic process.
- It involves executing of code as well as human based execution of program.
- It uses methods like black box and white box testing.
- It can catch errors that veri_cation cannot catch.

4. WORKING OF PROPOSED SYSTEM:

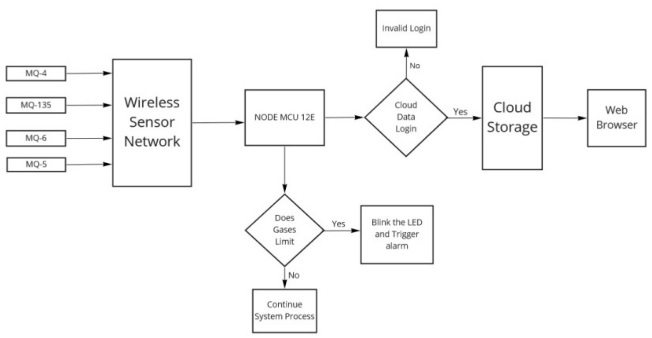


Figure 2: Architecture Diagram of Cloud Based Air Pollution Monitoring System

Sensor layer:

Sensor network used to detect the presence CO2, CH4, CO, NH3, C0, LPG, dust particle content in the air. The first sensor layer includes the electronic hard- ware circuits and the software components, both for sensor nodes and the gateway node, which assembled using a Node MCU with a wireless expansion module for cap- turing the data.

MQ-135 is sensitive to smoke, alcohol and CO2. MQ-5 is sensitive to LPG, hydrogen, carbon monoxide. MQ-4 is used to detect methane.MQ-6 detect the LPG, Butane. Libelium provides generic platforms capable of running, interfac- ing, and data collection from these sensors. The open platforms allow the users to customize the gases they desire to monitor in their facilities. These node moni- tors the sensors, acquires reading, stores the data on cloud and transmits to the gateway.

NodeMCU is an open source IoT platform. It includes firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems and hardware which is based on the ESP-12 module. The term "NodeMCU" by default refers to the firmware rather than the development kits. The firmware uses the Lua scripting language. It is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266. It uses many open source projects, such as lua-cjson and SPIFF [5]. The NodeMCU was able to support the MQTT IoT protocol, using Lua to access the MQTT broker. Another important update was made on 30 Jan 2015, when Devsaurus ported the u8glib to NodeMCU project, enabling NodeMCU to easily drive LCD, Screen, OLED, even VGA displays.

Application Layer:

Web services are design and implement using set of protocol and format that are used to process and store data on the cloud. Ubidots cloud platform is used to store the data. This information published on a Web page so that users would be able to access it through their Web browser. In order to send messages from the Gateway node to the application server.

Cilent layer:

This layer provide graphical user interface to visualize the information. This layer consist of Web graphical user interface, providing a visual information about environment parameter in order to allow the communication with the WSN and users.We used the ubidots cloud platform to visualize the data and display it on the web page. The client can access the data that is being displayed on the dash- board but the client will not be able to do any modification to the data received. We can install this system anywhere and can also trigger some device when pol- lution goes beyond some level,send alert to the user. The Web application pro- vides users with the online real time monitoring of the sensor data and also allows the retrieval and control of sensors remotely. Th GSM module is used to send the alert message to the administrator of the system.

4.1 Algorithm Deatils:

Step 1: Collect data from all the Gas sensors via Node MCU.

Step 2: Analyze the received data, display the received data on LCD dis- play. If quantity of pollutant in air arise blink the LED and trigger an alarm.

Step 3: Store the collected data in a cloud database to be used for backup.

Step 4: The collected data can be accessed through web browser from any- where using provided log-in credentials.

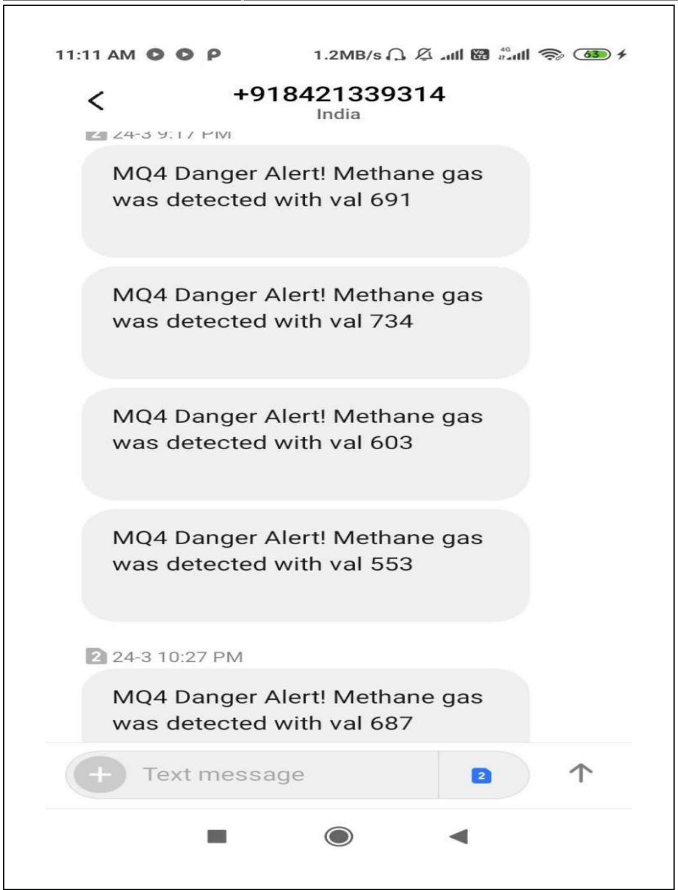
Step 5: For analysis purpose, graphs and charts are generated to calculate the percentage of pollutant in given area.

5. RESULT ANALYSIS:

1. To display the sensed data in user friendly format in LCD display panel. The LCD display continuously shows the real time values measured by gas sen- sors. This will provide a good solution for the analysis of environment.

2. The collected data by sensor node can be stored on cloud server that can be accessed through web browser from anywhere using provided log-in cre- dentials.

3. The alert msg when pollutants in air arises beyond threshold is send to sys- tem administrator.



6. CONCLUSION AND FUTURE WORK:

The proposed system provides low cost, low power, compact and highly accurate system for monitoring the environment with the dedicated sensors remotely from any place in this world. A perfect tradeoff between accuracy and cost is achieved by making use of single board minicomputer Raspberry pi and appropriate sensors leading to a well- grounded system. Datasheets available on the dashboard of IBM Bluemix account will help in framing good policies against the increasing level of pollution to ensure healthful environment. Air quality monitoring system can be more advantageous if pollutants like Sulfur dioxide, nitrogen dioxide, ground level ozone etc. are also monitored. Furthermore, long- term pollution patterns can be discovered and certain relationships between the air pollutants can be found.

REFERENCES:

I. Phala, Kgoputjo Simon Elvis, Anuj Kumar, and Gerhard P. Hancke. "Air quality monitoring system based on ISO/IEC/IEEE 21451 standards." IEEE Sensors Journal 16, no. 12, pp. 5037-5045, 2016.

II. Zheng, Kan, Shaohang Zhao, Zhe Yang, Xiong Xiong, and Wei Xiang. "Design and implementation of LPWA-based air quality monitoring system." IEEE Access 4, pp. 3238- 3245, 2016.

III. Marinov, Marin B., Ivan Topalov, Elitsa Gieva, and Georgi Nikolov, "Air quality monitoring in urban environments", 39th IEEE International Spring Seminar In Electronics Technology (ISSE), pp. 443-448, 2016.

IV. Liu, X., & Baiocchi, O. (2016, October) "A comparison of the definitions for smart sensors, smart objects and Things in IoT". 7th IEEE Conference In Information Technology, Electronics and Mobile Communication(IEMCON),pp. 1-4,2016..

V. Upton, Eben, and Gareth Halfacree. Raspberry Pi user guide. John Wiley & Sons, 2014.

VI. Shete, Rohini, and Sushma Agrawal. "IoT based urban climate monitoring using Raspberry Pi", IEEE International Conference In Communication and Signal Processing (ICCSP), pp. 2008- 2012, 2016.

VII. Jha, Mukesh, Prashanth Reddy Marpu, Chi-Kin Chau, and Peter Armstrong, "Design of sensor network for urban micro-climate monitoring", First IEEE International Conference In Smart Cities(ISC2), pp.1-4, 2015.

VIII. Nastic, Stefan, Sanjin Sehic, Duc-Hung Le,Hong- Linh Truong, and Schahram Dustdar, "Provisioning software-defined IoT cloud systems", International Conference In Future Internet of Things and Cloud (FiCloud), pp.288-295, 2014.

IX. www.greenpeace.org

X. Baralis, Elena, Tania Cerquitelli, Silvia Chiusano, Paolo Garza, and Mohammad Reza Kavoosifar, "Analyzing air pollution on the urban environment", 39th IEEE International Convention In Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 1464-1469, 2016.